Amendments to the Claims:

This listing of claims will replace all prior versions, and listing, of claims in the application.

1. (Currently amended) An overlapping command submitting method of dynamic cycle pipeline, for a chip having a pipeline including a plurality of stages, ~~wherein a command exits the pipeline at a predetermined stage without passing through stages subsequent to the predetermined stage in the pipeline,~~ comprising the following steps:

(a) reading a command from a command buffer and storing it in a command register;

(b) decoding the command;

(c) preprocessing operators of the command, preparing initial operators of each stage of the pipeline, and storing them into an initialization register;

(d) judging whether the pipeline is not full, if it is not full, directly inserting a new command; otherwise, waiting for an exiting signal from the command in the pipeline, ~~provided~~ the exiting signal being sent during a pipeline period immediately before a last cycle of the command exiting the pipeline;

(e) after receiving the exiting signal, judging whether there is command relevance between the new command to be inserted and an old command to exit, if yes, then inserting the new command after the old command exit; otherwise, performing a next step;

(f) when the old command is in the last cycle of the pipeline, submitting the new command to the pipeline;

wherein the new command and the old command each contain a field,

wherein step (e) includes determining whether there is any field conflict between the new command and the old command,

wherein if there is any field conflict between the new command and the old command, then a field branch is created, the field branch including a major current register for storing the field of the new command and a branch current register for storing the field of the old command, the field of the new command being added into the pipeline

when submitting, and the field of the old command being entered into the branch current register and maintained in the field branch until the old command uses the field for the last time,

wherein the major current register and the branch current register are connected to a hardware processing module through a multi-route switch,

wherein if the new command is processed by the hardware processing module, the hardware processing module receives an input of the major current register in which the field of the new command is stored, and if the old command is processed by the hardware processing module, the hardware processing module receives an input from the branch current register in which the field of the old command is stored, and

wherein if there is no field conflict, a field switch is conducted in a corresponding pipeline segment after submitting.

2. (Currently Amended) The command submitting method of Claim 1, wherein the Step (b) also includes a step of judging whether there is an illegal command, if there is, then deleting the illegal command and returning to Step (a); otherwise, conducting the next step.

3. (Currently Amended) The command submitting method of Claim 2, wherein said illegal command includes: ~~the instructions~~ a command with an incorrect command code and/or carrying unreasonable command parameters.

4. (Previously presented) The command submitting method of Claim 1, wherein the exiting signal is released two stages before the new command enters the pipeline.

5. (Previously presented) The command submitting method of Claim 1, wherein the command relevance means that the new command and the old command cannot share the hardware processing module in the same pipeline stage.

6. (Currently Amended) The command submitting method of Claim 1, wherein in the Step (e), it is also judged in which stage of the pipeline, a field switch shall be conducted for the new command and the old command[[s]], and the field switch is completed in a corresponding pipeline stage where the new command and the old command[[s]] overlaps.

7. (Canceled)

8. (Currently Amended) The command submitting method of Claim 1, wherein in the Step (c), it is required to provide [[the]] an initial status of [[the]] each command[[s]] ~~at the~~ when each command enters ~~entry to~~ the pipeline.

9. (Currently Amended) The command submitting method of Claim 1, wherein ~~the said~~ each command[[s]] includes reading/writing memory commands, reading/writing control register commands and various searching commands.

10. (Currently Amended) A chip on which the method according to Claim 1, is carried out having the dynamic cycle pipeline ~~structure~~, comprising: an interface of host computer, input buffer, command processing unit, and result unit; the command processing unit comprises: command interpreter and pipeline performing unit; characterized in that the command interpreter further comprises: command buffer controller[[s]], command register, operator processing unit ~~of operator~~, pipeline initialization register and control automaton, which are connected in order; the control automaton controls the command buffer controller to read a command from the command buffer, and stores the command into the command register; the control automaton decodes the command, and controls the operator processing unit ~~of operator~~ to prepare initial operators of each pipeline stage according to [[the]] a type of the command, and stores them into the pipeline initialization register.

11. (Previously presented) The command submitting method of Claim 2, wherein the exiting signal is released two stages before the new command enters the pipeline.

12. (Previously presented) The command submitting method of Claim 2, wherein the command relevance means that the new command and the old command cannot share the hardware processing module in the same pipeline stage.

13. (Currently Amended) A chip on which the method according to Claim 2, is carried out having the dynamic cycle pipeline ~~structure~~, comprising: an interface of host computer, input buffer, command processing unit, and result unit; the command processing unit comprises: command interpreter and pipeline performing unit; characterized in that the command interpreter further comprises: command buffer controller[[s]], command register, operator processing unit ~~of operator~~, pipeline initialization register and control automaton, which are connected in order; the control automaton controls the command buffer controller to read a command from the command buffer, and stores the command into the command register; the control automaton decodes the command, and controls the operator processing unit ~~of operator~~ to prepare initial operators of each pipeline stage according to [[the]] a type of the command, and stores them into the pipeline initialization register.

14. (Currently Amended) A chip on which the method according to Claim 3, is carried out having the dynamic cycle pipeline ~~structure~~, comprising: an interface of host computer, input buffer, command processing unit, and result unit; the command processing unit comprises: command interpreter and pipeline performing unit; characterized in that the command interpreter further comprises: command buffer controller[[s]], command register, operator processing unit ~~of operator~~, pipeline initialization register and control automaton, which are connected in order; the control automaton controls the command buffer controller to read a command from the command buffer, and stores the command into the command register; the control automaton decodes the command, and controls the operator processing unit ~~of operator~~ to prepare

initial operators of each pipeline stage according to [[the]] a type of the command, and stores them into the pipeline initialization register.

15. (Currently Amended) A chip on which the method according to Claim 6, is carried out having the dynamic cycle pipeline ~~structure~~, comprising: an interface of host computer, input buffer, command processing unit, and result unit; the command processing unit comprises: command interpreter and pipeline performing unit; characterized in that the command interpreter further comprises: command buffer controller[[s]], command register, operator processing unit ~~of operator~~, pipeline initialization register and control automaton, which are connected in order; the control automaton controls the command buffer controller to read a command from the command buffer, and stores the command into the command register; the control automaton decodes the command, and controls the operator processing unit ~~of operator~~ to prepare initial operators of each pipeline stage according to [[the]] a type of the command, and stores them into the pipeline initialization register.

16. (Currently Amended) A chip on which the method according to Claim 7, is carried out having the dynamic cycle pipeline ~~structure~~, comprising: an interface of host computer, input buffer, command processing unit, and result unit; the command processing unit comprises: command interpreter and pipeline performing unit; characterized in that the command interpreter further comprises: command buffer controller[[s]], command register, operator processing unit ~~of operator~~, pipeline initialization register and control automaton, which are connected in order; the control automaton controls the command buffer controller to read a command from the command buffer, and stores the command into the command register; the control automaton decodes the command, and controls the operator processing unit ~~of operator~~ to prepare initial operators of each pipeline stage according to [[the]] a type of the command, and stores them into the pipeline initialization register.

17. (Currently Amended) A chip on which the method according to Claim 8, is carried out having the dynamic cycle pipeline ~~structure~~, comprising: an interface of host computer, input buffer, command processing unit, and result unit; the command processing unit comprises: command interpreter and pipeline performing unit; characterized in that the command interpreter further comprises: command buffer controller[[s]], command register, operator processing unit ~~of operator~~, pipeline initialization register and control automaton, which are connected in order; the control automaton controls the command buffer controller to read a command from the command buffer, and stores the command into the command register; the control automaton decodes the command, and controls the operator processing unit ~~of operator~~ to prepare initial operators of each pipeline stage according to [[the]] a type of the command, and stores them into the pipeline initialization register.

18. (Currently Amended) A chip on which the method according to Claim 9, is carried out having the dynamic cycle pipeline ~~structure~~, comprising: an interface of host computer, input buffer, command processing unit, and result unit; the command processing unit comprises: command interpreter and pipeline performing unit; characterized in that the command interpreter further comprises: command buffer controller[[s]], command register, operator processing unit ~~of operator~~, pipeline initialization register and control automaton, which are connected in order; the control automaton controls the command buffer controller to read a command from the command buffer, and stores the command into the command register; the control automaton decodes the command, and controls the operator processing unit ~~of operator~~ to prepare initial operators of each pipeline stage according to [[the]] a type of the command, and stores them into the pipeline initialization register.